OIPE IAP47
JAN 2 2 2007
PATENT & TRADEMARK OFFICE

REMARKS

Reconsideration of the above-identified application in view of the following analysis and remarks is respectfully requested.

Claims 1 – 11 are pending in this case.

§103 Rejections

Claims 1 – 11 have been rejected under 35 USC §103 as being unpatentable over Tso [US 6,088,803] in view of Hall [US 2004/0054928]. The Applicants respectfully traverse the rejection, as detailed below.

It is well-appreciated that Tso discloses a virus-checking system which attempts to maximize efficient data transfer from server to client, by withholding a portion of a requested file during the streaming transfer of data, pending the confirmation that no virus was detected. [Tso column 2, lines 16 – 19; column 3, lines 11 – 14]

Tso, however, fails to disclose or reasonably suggest an envelope file, as defined in the present application, which is described *inter alia* in Figure 2 of the present application and the descriptive text thereof as (emphasis supplied):

> [Present application, paragraph 0032] *FIG. 2 schematically illustrates the parts of an envelope file, according to a preferred embodiment of the present invention. The envelope file **50** comprises:*
>
> [Present application, paragraph 0033] *an executable part **51**;*
>
> [Present application, paragraph 0034] *the original file **52**; and*
>
> [Present application, paragraph 0035] *an integrity indicator **53**.*
>
> [Present application, paragraph 0011] *setting the indicator on the envelope file to indicate the inspection result thereof...*

It is noted that the essential elements of an envelope file, as disclosed above, are discussed in detail within the present application's specification, and are recited as supported therein in claim 1.

Instead, Tso discloses that the file itself is sent by the server portion-by-portion, while withholding a portion (emphasis supplied):

> [Tso, col. 3 lines 14 – 16] ...*content server 7 will transmit a requested data object as a series of contiguous portions.*

> [Tso, col. 3 lines 24 – 26] ...*it always withholds a portion (or a segment of a portion) of the requested file most-recently received from content server 7...*

In addition, Tso fails to disclose an envelope file as "a copy of the object missing the final segment" (as suggested by the present Office Action), but rather makes a copy only for maintaining at the network device (emphasis supplied):

> [Tso, col. 3 lines 39 – 41] ...*as network device **4** is streaming the requested file to client device **12**, it maintains a working copy of the transmitted portions...*

Moreover, Tso fails to disclose an integrity indicator. The Office Action suggests that Tso holds "at least part of said envelope file which comprises the indicator (column 3 lines 23 – 28). However, the cited passage of Tso in fact fails to disclose such an indicator or setting such an indicator:

> [Tso, col. 3 lines 23 – 28] ...*network device **4** behaves differently from such known proxies in that it always withholds a portion (or a segment of a portion) of the requested file most-recently received from content server **7**, and does not transmit that withheld portion until at least another similarly-sized portion of the requested file is received."*

The above passage fails to disclose an integrity indicator of any sort and fails to disclose setting an integrity indicator, but rather deals only with issues involving

data transmission functions. In contrast, the specification of the present application defines such an indicator as indicating the integrity of the object, and discloses an operation of setting the indicator to indicate the results of the inspection (as cited above). These features are recited in claim 1.

Furthermore, the Office Action itself concedes that Tso's file "is not executable with code that extracts the object" (page 3). It is noted that code for extracting the object is an essential feature of an envelope file as defined in the present specification, *inter alia*, in Figure 2 and the accompanying description thereof, as detailed previously, and as recited in claim 1.

Accordingly, the Applicants respectfully maintain that Tso does not anticipate or reasonably suggest independent claim 1, and that claim 1 is therefore novel and non-obvious over Tso.

The Office Action, however, suggests that Hall supplies the features missing from Tso, specifically by suggesting that Hall [paragraph 0044] discloses "an executable wrapper used to protect files". This is given in the Office Action as the basis of the present 35 USC §103 rejection that Tso in view of Hall renders claim 1 as obvious. The Applicants respectfully traverse this rejection, and demonstrate below (emphasis supplied): (a) that Hall fails to disclose the features absent in Tso that are included as limitations in the present claims; and (b) that the nature of Hall is such that there is no reasonable motive or suggestion in the prior art for combining Hall with Tso, nor is there a reasonable expectation of success for any such combination. The cited paragraph of Hall reads as follows:

> [Hall, paragraph 0044] *Commands **400** are a second type of latent software on target server **22** that are altered to include authorization and notification algorithm **80**. An unaltered command **400** is immediately executed **402** once an invocation call **404** is received. As*

*shown in **FIGS. 7-10**, for commands executed from the command line, a "wrapper" script **506** can be written that replaces the original command **400** to implement authorization and notification algorithm **80**. Wrapper **506** first calls the secure query/response protocol software **80**. If this call returns success, then wrapper **506** will execute the normal command **402**, which will be hidden in a non-standard location **110**, as shown, for example, in **FIG. 10**.*

First of all, Hall fails to teach an executable wrapper, because Hall demonstrably relies on drawings showing a "wrapper script **506**" and a "non-standard location **110**" as an essential part of the disclosure. In fact, however, there is no "wrapper script **506**" appearing in any of Hall's drawings, nor is there a "non-standard location **110**" in any of Hall's drawings, either. It is noted that Hall further makes reference to a "**Fig. 9**" and a "**Fig. 10**", but none of these alleged drawings, nor the alleged "wrapper script **506**", nor the alleged "non-standard location **110**" appear in Hall's application as published; there are only 8 drawing sheets in the published reference, none of which include these alleged drawings or cited references. The Applicants therefore respectfully maintain that the cited Hall references, in particular Hall paragraph [0044], are ambiguous and fundamentally fail to disclose any usable teaching relevant to the present invention.

Despite the above-noted failures of Hall to teach an executable wrapper, however, it is possible to conjecture what Hall may have intended, by inference from the next paragraph (emphasis supplied):

[Hall, paragraph 0045] *Under Linux, the "ls" command is typically stored in a well-known standard location **112** on the file system **114**, such as ("/bin/ls") as shown, for example, in **FIG. 9**. During the implementation of a monitored command **500** the original "ls" command **116** is moved from its standard location ("/bin/ls") **112** to the hidden location (/Ax77z423/bin/ls") **110**, as shown, for example, in **FIG. 10**, and thus becomes a "hidden command". The wrapper for "ls", ls<wrapper> **118** is then installed in "/bin/ls" with a call to the hidden command "ls" **116** that is installed in "/Ax77z423/bin/ls" of file system **114**. The hiding is successful, because the attacker has no way*

> *of practically guessing the pathname to locate the original commands* ***402****. All directory listing commands will be wrapped, so an attacker cannot execute such a listing without causing a notification, which will then detect and prevent further intrusion.*

It may be inferred from the above passage (in combination with the preceding paragraph 0044) that Hall's intention is to describe a script that replaces a native operating system command (such as the Linux *ls* command for listing a file system directory) for the purpose of detecting invasive attacks on the system. The script would invoke the native operating system command, so that an attacker who enters the command on the command line would immediately see the effect (such as the displayed directory listing for an *ls* command) and would presume that the system is functioning normally. The script, however, would also provide for other actions of which the attacker would not be aware — such as a notification of the intrusion. For this purpose, the script would given the same name as the native command, and would be put into the original location of the native command; the native command which the script invokes would have been previously moved to a location that could not easily be found by the attacker.

Such a "wrapper" script is entirely different from the envelope file of the present invention — it is different not only in purpose but also in structure and function. Structurally, for example, the envelope file of the present invention contains the original file code and data, and executable code to extract the original file, so that when the envelope file is executed, the original file is extracted — Hall fails to disclose a "wrapper script" with any of these features. At most, Hall's script would contain only a call to the original operating system command, but not the code for the command itself. Hall moreover fails to disclose a "wrapper script" that extracts anything. At most, Hall's script would only execute the original operating system

command. In addition, the motivation behind Hall's "wrapper script" would be to deceive an attacker and provide notification of an attack, whereas the motivation for the present invention's envelope file is to encapsulate the original file for improved transmission efficiency and increased safety in the event the original file is found to have malicious content, and also to provide a settable indicator of the data object's integrity. Furthermore, Hall fails to disclose a "wrapper script" for use in data transmission, nor would Hall's script be capable of aiding data transmission. In particular, Hall fails to disclose a "wrapper script" that is capable of encapsulating or containing arbitrary data objects from a server, and Hall also fails to disclose an integrity indicator. Thus, *Hall's concept of a "wrapper script" cannot be used with Tso's procedures for inspection and transmission of data objects*. Not only would there be no reasonable motivation for combining Tso and Hall, but there is no expectation of success in such a combination.

Once again, however, it is emphasized that Hall's failure to present a usable, unambiguous disclosure, as previously shown, precludes the use of Hall as a credible prior-art reference in this case.

Consequently, because of the above-detailed failures of Hall and Tso there is no *prima facie* case of obviousness against the present claims: (1) there is no motivation or suggestion in the prior art to combine the reference teachings of Tso and Hall; (2) there is no reasonable expectation of success from combining these references; and (3) the references when combined do not teach or suggest all the limitations of claim 1. Nor do these references anticipate or reasonably suggest any of the remaining claims of the present application, all of which depend from claim 1.

§112 Rejection

Claim 3 has been rejected under 35 USC §112 as being indefinite for failing to particularly point out and distinctly claim the subject matter which the Applicants regard as the invention. Specifically, the Office Action (page 2) states that:

> 3. Claim 3 recites the limitation "said code is auto-executable". The specification does not define the term "auto-executable".

The Applicants respectfully traverse this rejection, noting that the term "auto-executable", which is supported in the specification of the present application [US 2005/0235160, paragraph 0012], is a term of art in common usage, and that this term does indeed serve to distinctly point out and claim the subject matter which the Applicants regard as the invention. As supporting evidence for this, the Applicants note that in section 3, page 2 of the Office Action, it is stated that:

> ...the examiner is understanding this claim to mean that the object is automatically removed from the envelope when inspection and release is complete.

The Examiner is correct to apply the above understanding to claim 3, and the Applicants therefore respectfully maintain that claim 3, as readily-understood by a person ordinarily-skilled in the art, is not indefinite, and is thus in full compliance with 35 USC §112.

Conclusion

After careful and thorough consideration of the present Office Action, it has been shown that the cited prior art fails to anticipate or reasonably suggest the present claims; and furthermore, that the terms used in the claims, as supported by the specification, are not indefinite, but do in fact particularly point out and distinctly

claim the subject matter which the Applicants regard as the invention. In view of the above analysis and remarks it is respectfully submitted that the claims are indeed in accordance with 35 USC §103 and 35 USC §112, and are in condition for allowance. Accordingly, a notice of allowance is respectfully and earnestly solicited.

Respectfully submitted,

Mark M. Friedman
Attorney for Applicant
Registration No. 33,883

Date: January 14, 2007